

CLAIMS

What is claimed is:

1. A server system comprising:

5 one or more computers; and

a multi-layer application executing on the computers to handle client requests submitted by various client devices, the multi-layer application comprising:

10 a problem-solving logic layer to process the client requests according to an associated problem domain, the problem-solving logic layer containing one or more execution models to perform various sets of tasks when processing the client requests, the problem-solving logic layer producing replies to the client requests;

15 an execution environment layer to receive the client requests and select an appropriate execution model in the problem-solving logic layer for processing the client requests; and

an interfacing layer to interface the problem-solving logic layer with one or more resources so that the execution models may utilize the resources when processing the client requests; and

20 a presentation layer to receive the replies produced by the problem-solving logic layer and to structure the replies in a manner that makes the replies presentable on the various client devices.

2. A server system as recited in claim 1, wherein the execution

25 environment layer comprises a framework to receive the client requests and route the requests to the problem-solving logic for processing.

3. A server system as recited in claim 2, wherein the execution environment layer comprises one or more adapters to interface the framework with different types of the client devices.

5 4. A server system as recited in claim 1, wherein the execution environment layer comprises:

a model dispatcher to route the client requests to selected execution models in the problem-solving logic layer; and

10 a request dispatcher to structure the replies for return to the client devices.

15 5. A server system as recited in claim 1, wherein the multi-layer application can be adapted to receive requests from new client devices with incompatible communication protocols by substituting a new execution environment layer that supports the new client devices.

6. A server system as recited in claim 1, wherein one of the execution models is embodied as a set of discrete program modules, each program module performing a specific task.

20 7. A server system as recited in claim 1, wherein one of the execution models is embodied as an interaction-based model in which computer programs are defined by a series of interaction definitions.

25 8. A server system as recited in claim 1, wherein the execution models are embodied according to at least one of a command model, an action-view model, and a use case model.

9. A server system as recited in claim 1, wherein one of the execution models performs tasks according to a first business purpose, and the multi-layer application being reconfigurable to achieve a different business purpose by installing another execution model that performs tasks according to the second business purpose.

10. A server system as recited in claim 1, wherein the interfacing layer comprises:

a data abstraction layer to obtain data from the resources and map the data into a domain framework that models information flow for a specific problem domain; and

a data coordination layer that provides an interface for the problem-solving logic layer to access the domain framework of the data abstraction layer and obtain the data.

11. A server system as recited in claim 10, wherein the data coordination layer comprises one or more application data managers that interface the domain framework in the data abstraction layer into an application solution space of the problem-solving logic layer.

12. A server system as recited in claim 1, wherein the multi-layer application can be adapted to access new resources by substituting in a new interfacing layer that supports the new resources.

13. A server system as recited in claim 1, wherein the client devices support different data formats, the presentation layer being configured to select appropriate data formats for encoding the replies.

14. A server system as recited in claim 1, wherein the client devices support different communication protocols, the presentation layer being configured to select appropriate communication protocols for delivering the replies to the clients.

5

15. A server system as recited in claim 1, wherein the presentation layer is configured to determine how to display the replies for a particular client.

10

16. A server system as recited in claim 1, wherein the presentation tier is configured to determine at least one of (1) a layout of individual replies, (2) display attributes in which to present the replies, and (3) a presentation theme.

15

17. A server system as recited in claim 1, wherein the presentation layer comprises:

a presentation module to determine how the replies will appear on the client devices to users; and

20

a rendering module, separate from the presentation module, to determine how to render the replies on the client devices.

25

18. A server system as recited in claim 1, further comprising an authentication module to authenticate the client devices or users of the client devices.

19. A server system as recited in claim 1, further comprising a constraint system to constrain operation of the multi-layer application according to a hierarchy of different constraints.

20. A server system as recited in claim 1, further comprising a constraint system to constrain operation of the multi-layer application according to multiple different constraints, the constraint system comprising a hierarchy of constraint layers, with each constraint layer containing a set of one or more constraints that customize operation of the multi-layer application.

21. A server system as recited in claim 1, further comprising:
a constraint hierarchy of multiple constraint layers, each constraint layer containing a set of one or more constraints that constrain operation of the multi-layer application, the constraint layers being organized within the constraint hierarchy such that a first constraint layer limits a second constraint layer but the second constraint layer does not limit the first constraint layer; and

a constraint resolver to resolve the constraint layers so that operation of the multi-layer application is constrained by a set of the constraints in the constraint layers.

22. A server system as recited in claim 21, wherein the hierarchy of constraints comprises constraints selected from a group of constraints comprising:

legally mandated constraints to constrain operation of the multi-layer application according to legal principles;

company-mandated constraints to constrain operation of the multi-layer application according to preferences of a company that operates the application;

customer constraints to constrain operation of the multi-layer application according to preferences of customers;

cultural constraints to constrain operation of the multi-layer application according to cultural aspects; and

5 end user constraints to constrain operation of the multi-layer application according to preferences of an end user.

10 **23.** A server system as recited in claim 1, further comprising a security policy enforcement module to enforce security restrictions on accessing information stored at the one or more resources.

15 **24.** A server system as recited in claim 23, wherein the security policy enforcement module is to enforce the security restrictions based on a set of low-level security rules defined using high-level permission concepts.

25. A server system as recited in claim 1, wherein the presentation layer includes a form processor to generate a data input form for the multi-layer application by automatically adding, to a form definition that defines the data input form, validation code to validate subsequent inputs to one or more fields of the data input form.

26. A server system as recited in claim 25, wherein the form processor is to generate the data input form by identifying one or more custom tags associated with the data input form, to replace each of the one or more custom tags with another tag, and further to add to the form definition, for each of the one or more replaced tags, validation code to validate subsequent inputs to a field corresponding to the tag.

27. A server system as recited in claim 25, wherein the form processor is further to automatically identify one or more data input fields to be included in the form definition.

5 28. A server system as recited in claim 25, wherein the form processor is further to automatically identify one or more restrictions associated with a data input field of the data input form, and to determine the validation code based at least in part on the one or more restrictions.

10 29. A server system as recited in claim 1, further comprising:
a resource bundle containing locale-specific content that is authored for a particular locale; and
a resource bundle manager to populate a locale-independent core with the locale-sensitive content in the resource bundle to produce a computer-servable document that can be served by the multi-layer application to the particular locale.

15 30. A server system as recited in claim 29, wherein the resource bundle manager resides in the interfacing layer.

20 31. A business-oriented computer software architecture stored on one or more computer-readable media, comprising:

a framework module to receive client requests from different client devices;

25 a first business logic module to process the client requests received by the framework according to an associated business purpose, the first business logic module generating replies corresponding to the client requests;

a presentation module to structure the replies produced by the first business logic module in a manner that makes the replies presentable on the client devices; and

the business-oriented computer software architecture being reconfigurable to another business purpose by substituting a second business logic module for the first business logic module.

32. A business-oriented computer software architecture as recited in claim 31, wherein the framework module forms a container for the first and second business logic modules that allows the first and second business logic modules to be selectively added or removed from the architecture.

33. A business-oriented computer software architecture as recited in claim 31, wherein the first and second business logic modules are implemented as discrete programs, each program performing a specific task.

34. A business-oriented computer software architecture as recited in claim 31, wherein the client devices support different data formats, the presentation layer being configured to select appropriate data formats for encoding the replies.

35. A business-oriented computer software architecture as recited in claim 31, wherein the client devices support different communication protocols, the presentation layer being configured to select appropriate communication protocols for delivering the replies to the clients.

36. A business-oriented computer software architecture as recited in claim 31, wherein the presentation module comprises:

a presentation component to structure how the replies will appear; and
a rendering component, separate from the presentation component, to

5 configure how the replies are output on a particular client.

37. A business-oriented computer software architecture as recited in claim 31, wherein the presentation module generates a data input form by automatically adding, to a form definition that defines the data input form,
10 validation code to validate subsequent inputs to one or more fields of the data input form.

38. A business-oriented computer software architecture as recited in claim 37, wherein the presentation module automatically identifies one or more
15 data input fields to be included in the form definition.

39. A business-oriented computer software architecture as recited in claim 37, wherein the presentation module automatically identifies one or more restrictions associated with a data input field of the data input form and
20 determines the validation code based at least in part on the one or more restrictions.

40. A business-oriented computer software architecture as recited in claim 31, further comprising a domain framework to model information flow
25 for a particular business domain.

41. A business-oriented computer software architecture as recited in claim 31, further comprising a constraint hierarchy with a hierarchical set of constraints that specify how the replies should be structured to customize the replies.

42. A business-oriented computer software architecture as recited in claim 31, further comprising:

a resource bundle containing locale-specific content that is authored for a particular locale; and

a resource bundle manager to populate a locale-independent core with the locale-sensitive content in the resource bundle to produce the replies.

43. A business-oriented computer software architecture stored on one or more computer-readable media, comprising:

a first layer to obtain data from external resources and map the data into a domain framework that models information flow for a specific problem domain;

a second layer to provide an interface into the domain framework of the first layer;

a third layer to process requests and produce replies, the third layer utilizing the second layer to access the domain framework and retrieve the data obtained by the first layer from the external resources;

a fourth layer to structure the replies for presentation and rendering on diverse client devices; and

wherein anyone of the first layer, the second layer, the third layer, and the fourth layer may be changed independently of other layers to modify operation of the computer software architecture.

44. A business-oriented computer software architecture as recited in claim 43, wherein the first layer may be updated to obtain data from new external resources.

45. A business-oriented computer software architecture as recited in claim 43, wherein the third layer is an original third layer that processes the requests according to a first business domain, further comprising a new third layer substituted for the original third layer to process the requests according to a second business domain.

46. A business-oriented computer software architecture as recited in claim 43, wherein the fourth layer can be updated to structure the replies for new client devices.

47. A business-oriented computer software architecture as recited in claim 43, wherein the fourth layer can be modified to structure the replies for presentation to users in different regions of the world.

48. A method comprising:
receiving requests from multiple clients, the requests being related to a problem domain;
processing the requests within problem-solving logic to produce replies within the problem domain, the processing comprising requesting data to be used in formulating the replies;
retrieving the data from one or more external resources and mapping the data to a domain framework for the problem domain, the domain framework being independent from the problem-solving logic; and

interfacing the problem-solving logic to the domain framework to obtain the data for use in processing the request.

49. A method as recited in claim 48, further comprising structuring the replies for presentation to the clients.

50. A method as recited in claim 48, further comprising:
structuring the replies to define how the replies will appear when presented at the clients; and

independent of said structuring, conforming the replies to output capabilities of the clients.

51. A method as recited in claim 48, further comprising constraining how the replies are presented according to a hierarchy of constraints, wherein the hierarchy of constraints comprises multiple constraints such that a first constraint limits a second constraint but the second constraint does not limit the first constraint.

52. A method for creating server applications for multiple different problem domains, comprising:

providing a framework to receive client requests from multiple different clients, individual clients communicating messages using different protocols and formats;

providing a resource structure to access resources that provide data related to the different problem domains;

creating a first server application for a first problem domain by interfacing first domain logic with the framework and the resource structure,

the first domain logic being configured to process the client requests for the first problem domain;

creating a second server application for a second problem domain by interfacing second domain logic with the framework and the resource structure,
5 the second domain logic being configured to process the client requests for the second problem domain; and

the first and second domain logic producing replies that the framework returns to the clients using the protocols and messaging formats employed by the clients.

10 **53.** A method as recited in claim 52, wherein the providing a resource structure comprises providing a resource structure that retrieves data from one or more external resources and maps the data to a domain framework that models information flow for a specific problem domain.

15 **54.** A method as recited in claim 52, further comprising:
structuring the replies to define how the replies will appear when presented at the clients; and

independent of said structuring, conforming the replies to output
20 capabilities of the clients.

55. A method as recited in claim 52, further comprising generating a data input form for service to a client by automatically adding, to a form definition that defines the data input form, validation code to validate
25 subsequent inputs to one or more fields of the data input form.

56. A method as recited in claim 52, further comprising automatically identifying one or more data input fields to be included in a data input form based on the first and second domain logic.

5 **57.** A method as recited in claim 52, further comprising forming a reply for a particular locale by retrieving a locale-independent core that contains locale-independent elements and populating the locale-independent core with locale-sensitive content that is appropriate for the particular locale.

10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000